

IN THE UNITED STATES DISTRICT COURT
FOR THE NORTHERN DISTRICT OF CALIFORNIA

IMPLICIT NETWORKS INC,

Nos. C10-3365 SI; C 10-4234 SI

Plaintiff,

v.

F5 NETWORKS INC,

Defendant.

**ORDER GRANTING DEFENDANTS'
MOTIONS FOR SUMMARY JUDGMENT**

IMPLICIT NETWORKS INC,

Plaintiff,

v.

JUNIPER NETWORKS, INC.

Defendant.

Currently before the Court are summary judgment motions by defendant Juniper Networks (Case No. 10-4234 SI) and defendant F5 Networks (Case No. 10-3365 SI) for non-infringement; and a motion for summary judgment by defendant Juniper Networks, in which defendant F5 Networks joins, as to invalidity of the asserted patents. Having considered the arguments of counsel and the papers submitted, the Court GRANTS defendants' motions.

BACKGROUND

In these two related cases, plaintiff Implicit Networks, Inc. accuses defendants' products of infringing two patents owned by plaintiff: U.S. Patent No. 6,629,163, as issued September 30, 2003 and as it emerged after reexamination on June 22, 2010 ('163 Patent); and U.S. Patent No. 7,711,857 ('857

1 Patent), issued May 4, 2010 as a continuation application from ‘163. In early 2012, Juniper filed
2 requests for *inter partes* reexamination on both patents. The *inter partes* proceedings are still pending,
3 although the examiners have issued Patent Action Closing Prosecutions (ACPs) concluding that both
4 patents are invalid as anticipated and/or obvious.

5 According to Implicit, the patents cover a modular networking system which dynamically selects
6 software routines (“modules” or “beads”) after the arrival of the first packet of a message to build a data
7 processing path for the subsequent packets of the message (“flow”). See, e.g., Implicit’s Opposition to
8 Motion for Summary Judgment on Invalidity (“Invalidity Oppo.”) at 4-7. Since Implicit’s system
9 creates the processing path dynamically and only after the arrival of the first packet, Implicit’s system
10 is flexible and can be changed to accommodate different processing steps, can be adapted to handle
11 different sorts of technology/flows, and can adopt “policies” based on administrator direction. *Id.*

12 Implicit accuses two lines of products made by defendant Juniper Networks, Inc.: the SRX and
13 J series of gateways and routers that use “flow based” processing of internet traffic. Implicit accuses
14 F5’s “BIG-IP” IP networking products which act as intermediaries and transfer data between users (e.g.,
15 a consumer) and servers (e.g., an e-commerce business like Amazon). The BIG-IP products help
16 “enterprise applications” sort and get traffic to the appropriate servers.

17 18 DISCUSSION

19 I. Defendants’ Motion on Invalidity

20 Juniper, joined by F5, moves for summary judgment on invalidity, arguing that the asserted
21 claims are invalid as disclosed or rendered obvious by Daniel Decasper, *et al.*, “Router Plugins: A
22 Software Architecture for Next Generation Routers,” Computer Communication Review, a publication
23 of ACM SIGCOMM, Vol. 28, No. 4 Oct. 1998. (Decasper98). Juniper also argues that the asserted
24 claims are rendered obvious by Decasper98 in combination with IBM, Local Area Network Products
25 Concepts and Products: Routers and Gateways (May 1996) (IBM96) and Mark Nelson and Jean Loup
26 Gailly, the Data Compression Book, M&T Books (2nd ed. 1996) (Nelson).

27 Juniper notes that the PTO examiners in both of the pending *inter partes* reexamination
28 proceedings have rejected all of the asserted claims. The PTO issued an Action Closing Prosecution

(ACP) in the ‘163 Patent reexam on October 1, 2012 and issued an ACP in the ‘857 Patent reexam on December 21, 2012. *See* ‘163 Patent ACP, Declaration of Douglas Dixon [Dixon Decl., Docket No. 167-5], Ex 15; ‘857 Patent ACP, Docket No. 201. The two panels of examiners found that Claims 1, 5, and 35 in the ‘163 Patent and Claims 1, 4 and 10 of the ‘857 Patent were anticipated in light of Decasper98 and obvious in light of Decasper98 alone and/or in combination with IBM96 and Nelson.¹ Implicit contends that the PTO actions are irrelevant for two reasons. First, the PTO examiners applied only a “preponderance of the evidence” standard of review in determining invalidity, but this Court must apply the higher “clear and convincing” standard. Second, Implicit notes that the *inter partes* proceedings are far from over. Following the issuance of the ACPs, patent owners are allowed to submit additional comments to the examiners and an appeal may follow. *See, e.g.*, Manual of Patent Examining Procedures, §§2671, 2672.

As relevant to the motion for summary judgment on invalidity, the parties’ dispute centers on five claim limitations, designated by the parties “1a,” “1b,” “1e,” “1f” and “1g” and shown in bold below, which are all present in Claim 1 of the ‘163 Patent:

Claim 1 of ‘163 Patent:

1. A method in a computer system for processing a message having a sequence of packets, the method comprising:
 - [1a] providing a plurality of components, each component being a software routine for converting data with an input format into data with an output format;**
 - [1b] for the first packet of the message,**
dynamically identifying a non-predefined sequence of components for processing the packets of the message such that the output format of the components of the non-predefined sequence match the input format of the next component in the non-predefined sequence, wherein dynamically identifying includes selecting individual components to create the non-predefined sequence of components after the first packet is received; and
storing an indication of each of the identified components so that the non-predefined sequence does not need to be re-identified for subsequent packets of the message; and
 - for each of a plurality of packets of the message in sequence,
 - [1e] for each of a plurality of components in the identified non-predefined sequence,**
 - retrieving state information relating to performing the processing of the component with the previous packet of the message;**
 - [1f] performing the processing of the identified component with the**

¹ For the ‘163 Patent, the PTO also found that the claims were anticipated and/or obvious in light of Patent No. 6,243,667 (Kerr). *See generally* Dixon Decl., Ex. 15. However, the PTO rejected that same argument with respect to Claims 1, 4, and 10 in the ‘857 Patent ACP. ‘857 ACP at 35.

packet and the retrieved state information; and storing state information relating to the processing of the component [1g] with packet for use when processing the next packet of the message.

See Dixon Decl., Ex 15 (italics added in reexamination; bold added by Court to show limitations involved in challenge to validity); *see also* Dixon Decl., Ex. 19 (Report of Dr. Scott Nettles on Infringement by Juniper; identifying limitations within claims).²

A. Legal Standard

i. In General

A patent is presumed valid after the PTO examination process, based on “the basic proposition that a government agency such as the then Patent Office was presumed to do its job.” *Am. Hoist & Derrick Co. v. Sowa & Sons, Inc.*, 725 F.2d 1350, 1359 (Fed. Cir. 1984) (*abrogated on other grounds* by *Therasense, Inc. v. Becton, Dickinson & Co.*, 649 F.3d 1276 (Fed. Cir. 2011)) (citing *Morgan v. Daniels*, 153 U.S. 120 (1894)). The defendant carries a high burden on summary judgment of invalidity, as the “moving party seeking to invalidate a patent at summary judgment must submit such clear and convincing evidence of invalidity so that no reasonable jury could find otherwise.” *Eli Lilly & Co. v. Barr. Labs*, 251 F. 3d 955, 962 (Fed. Cir. 2001). The presumption of validity can nonetheless be overcome with sufficient evidence. *See Magnivision, Inc. v. Bonneau Co.*, 115 F.3d 956, 960 (Fed. Cir. 1997) (“The validity of a patent is always subject to plenary challenge on its merits. A court may invalidate a patent on any substantive ground, whether or not that ground was considered by the patent

² Juniper argues that the elements of the other asserted claims *not* corresponding to the five limitations identified by Implicit (*i.e.*, non-contested elements of claims 1, 15 and 35 of the ‘163 Patent and non-contested elements of Claims 1, 4, and 10 of the ‘857 Patent) are disclosed by Decasper98 and, therefore, at a minimum partial summary judgment should be entered in Juniper’s favor on those claims/elements. *See* Juniper Motion re Invalidity at 19-21. Implicit does not contest this in its Opposition. In Reply, Juniper also argues that because Implicit only challenges whether Decasper98 covers sub-aspects of each of the limitations identified in Claim 1 (*e.g.*, Implicit does not dispute that Decasper98 discloses “a plurality of components, each component being a software routine” from 1a) summary judgment should be granted in Juniper’s favor on the “undisputed aspects” of elements 1a, 1b, 1e, 1f and 1g. Juniper Reply on Invalidity at 2; *see also* Juniper’s Slides from Summary Judgment Hearing at 4-8 (identifying undisputed claim elements).

1 examiner.”).³

2 The moving party’s burden is “especially difficult” when the prior art references presented were
3 considered by the patent examiner during prosecution. *Glaxo Group Ltd. v. Apotex, Inc.*, 376 F.3d 1339,
4 1348 (Fed. Cir. 2004). But when, as here, additional evidence is presented by the moving party, “the
5 burden may be more or less easily carried because of the additional evidence.” *Applied Materials, Inc.*
6 *v. Advanced Semiconductor Materials Am., Inc.*, 98 F.3d 1563, 1569 (Fed. Cir. 1996). New evidence
7 supporting an invalidity contention may “carry more weight” than evidence previously considered by
8 the PTO. *Microsoft Corp. v. i4i Ltd. P’ship*, 131 S.Ct. 2238, 2251 (2011).⁴ As the Supreme Court has
9 held, “[s]imply put, if the PTO did not have all material facts before it, its considered judgment may lose
10 significant force. . . . And, concomitantly, the challenger’s burden to persuade the jury of its invalidity
11 defense by clear and convincing evidence may be easier to sustain.” *Id.* (internal citations omitted).

12 13 **ii. Anticipation**

14 Under 35 U.S.C. § 102(b):

15 A person shall be entitled to a patent unless—

16 * * *

17 (b) the invention was patented or described in a printed publication in this or a foreign country or
18 in public use or on sale in this country, more than one year prior to the date of the application for
19 patent in the United States

20 In determining validity of a patent claim over the prior art, a two-step process applies. The first
21 step is the claim construction by the Court. *See Smiths Indus. Med. Sys., Inc. v. Vital Signs, Inc.*, 183
22 F.3d 1347, 1353 (Fed. Cir. 1999). The second step is a comparison of the asserted claims against the
23 prior art reference. A determination that a claim is invalid for anticipation requires a finding that “each
24 and every limitation is found either expressly or inherently in a single prior art reference.” *Celeritas*

25 ³ *See also Interconnect Planning Corp. v. Feil*, 774 F.2d 1132, 1139 (Fed. Cir. 1985) (“[t]he
26 Examiner’s decision, on an original or reissue application, is never binding on the court. It is, however,
27 evidence the court must consider in determining whether the party asserting invalidity has met its
28 statutory burden by clear and convincing evidence.”).

⁴ Citing *SIBIA Neurosciences, Inc. v. Cadus Pharmaceutical Corp.*, 225 F.3d 1349, 1355-1356
(Fed.Cir. 2000) (“[T]he alleged infringer’s burden may be more easily carried because of th[e]
additional [evidence]”); *Group One, Ltd. v. Hallmark Cards, Inc.*, 407 F.3d 1297, 1306 (Fed. Cir. 2005)
(similar).

Techs. Inc. v. Rockwell Int'l Corp., 150 F.3d 1354, 1360, 47 USPQ2d 1516, 1522 (Fed. Cir. 1998). The Federal Circuit has held that “it is axiomatic that that which would literally infringe if later anticipates if earlier.” *Bristol-Myers Squibb Co. v. Ben Venue Laboratories, Inc.*, 246 F.3d 1368, 1378 (Fed. Cir. 2001); *see also Lewmar Marine, Inc. v. Bariant, Inc.*, 827 F.2d 744, 747 (Fed. Cir. 1987).⁵

iii. Obviousness

Defendants must prove by clear and convincing evidence that “the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains.” 35 U.S.C. § 103(a); *KSR Int'l Co. v. Teleflex Inc.*, 550 U.S. 398, 406 (2007). Obviousness under 35 U.S.C. § 103 is a question of law, with underlying factual considerations regarding (1) the scope and content of the prior art, (2) the differences between the prior art and the claimed invention, (3) the level of ordinary skill in the art, and (4) any relevant secondary considerations. *Graham v. John Deere Co. of Kansas City*, 383 U.S. 1, 17-18 (1966); *see also Brown & Williamson Tobacco Corp. v. Philip Morris, Inc.*, 229 F.3d 1120, 1124 (Fed. Cir. 2000). A claimed invention is invalid for obviousness “if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains.” 35 U.S.C. § 103.⁶

“Although it is well settled that the ultimate determination of obviousness is a question of law, it is also well understood that there are factual issues underlying the ultimate obviousness decision.” *McGinley v. Franklin Sports, Inc.*, 262 F.3d 1339, 1349 (Fed. Cir. 2001). Summary judgment may be

⁵ Implicit does not contest that Decasper98, IBM96 and Nelson are publications under 35 U.S.C. § 102(b) which could qualify as prior art.

⁶ The parties’ experts describe a “person of ordinary skill in the art” differently. Juniper’s expert (Dr. Calvert) describes the person as one with a bachelor’s degree in a relevant field and at least four years’ work experience or a master’s degree in a relevant field and two years experience. *See* Declaration of Dr. Kenneth I. Calvert [Docket No. 167-1] and Exhibit 2 [Calvert Invalidity Expert Report], ¶ 42. Implicit’s expert describes the person as having a bachelor’s degree in the relevant field and at least two years’ experience. Expert Report of Dr. Scott Nettles on Validity Rebuttal [Docket No. 167-9], ¶ 30. Implicit points out the distinction, but does not argue that the distinction matters for resolution of the arguments made in the invalidity motion.

appropriate if “the content of the prior art, the scope of the patent claim, and the level of ordinary skill in the art are not in material dispute, and the obviousness of the claim is apparent in light of these factors.” *KSR*, 550 U.S. at 427 (citing *Graham*, 383 U.S. at 17). However, a factual dispute as to any one of these elements will defeat the motion. *Helifix Ltd. v. Blok-Lok, Ltd.*, 208 F.3d 1339, 1346 (Fed. Cir. 2000).

B. Whether Decasper98 Anticipates or Renders Obvious the Asserted Claims

Juniper’s expert, Dr. Kenneth L. Calvert, concludes that Decasper98 alone anticipates or renders obvious claims 1, 15, 35 of the ‘163 Patent as well as claims 1, 4 and 10 of the ‘857 Patent. *See* Calvert Declaration [Docket No. 167-1] and Exhibit 2 [Calvert Invalidity Expert Report] ¶¶ 543-611.⁷ Decasper98 describes a general “framework” for IP processing, that is “extensible” and can be applied to applications other than IP processing. Decasper98 (Dixon Ex. 1) at 2. Dr. Calvert argues Decasper98 describes a router system for processing message flows comprising multiple packets. Calvert Invalidity Report, ¶¶ 544-545. In Decasper98, packets traveling through the system encounter “gates” and at that point the system dynamically loads “plug-ins” (software routines) to process the message flow. *See* Decasper98 at 1,2, 4; *see also* Calvert Invalidity Report ¶¶ 546, 548. Decasper98 uses an “identification unit” which stores information relating to processing of particular types of packets, as well as “pointers to state information for each plugin that uses state information.” Calvert Invalidity Report, ¶549. The system maintains two types of state, including a “pointer” to the particular plugin and a pointer to “private data” for the plugin instance associated with the particular packet within a flow. *Id.*, ¶ 557. Based on those and other conclusions not contested by Implicit, Dr. Calvert opines that Decasper98 anticipates the asserted claims of the patents at issue. *Id.*, ¶¶ 543, 567.⁸

The PTO examiners in both the ‘163 and ‘857 *inter partes* reexams have likewise concluded that the patents in suit are anticipated and/or rendered obvious in light of Decasper98. *See* ‘163 Patent ACP

⁷ As discussed below, Dr. Calvert also finds that Decasper98 in view of IBM96 alone and with Nelson likewise render the asserted claims obvious. Calvert Invalidity Report ¶¶ 673-700.

⁸ The Court notes that Decasper98 also expressly contemplates use of the framework in an edge-router. Decasper98 at 2.

at 93,118 [Dixon Decl., Ex 15]; ‘857 Patent ACP at 7, 19.⁹ In the ‘163 Patent ACP, the examiner found that Decasper98 used “‘**plug-ins**’ as the claimed ‘**individual components**’ that are dynamically selected to create a sequence in accordance with this claim limitation.” ‘163 Patent ACP at 11, 20 (emphasis in original). The examiner also found that simply because Decasper98 might be designed to handle one type of traffic (IP), did not mean it could not read on the format-matching or compatibility limitation of the claims. *Id.*, 21-22. Finally, the examiner found that Decasper98 teaches the maintenance of message-specific state information as required by the claims. *Id.*, at 36; *see also id.* 118-137 [rejecting Claims 15 and 35 as anticipated by Decasper98].¹⁰ In the ‘857 ACP, the examiner found Claims 1, 4 and 10 anticipated by Decasper98. ‘857 ACP at 7.¹¹ In particular, the examiner found that even assuming Decasper98 uses a single IP format, Decasper98 still discloses the input and output matching feature of the claims. ‘857 ACP at 16. The examiner also found that Decasper98 discloses storing state information for each of the plug-in sequences/software routines to be used in processing the next packets as recited in the challenged claims. *Id.*, at 18-19.

In opposition, Implicit argues that Decasper98 cannot anticipate the claim limitations of the ‘163 patent identified above because Decasper98 describes a simple “one format fast router” which routes only IP packets. As Decasper98 handles only one type of format of message (IP),¹² it cannot cover the demultiplexing (conversion) processing claimed by the patents in suit, nor does it necessarily have to maintain state information for the processing of all packets in the message/flow. As such, Implicit argues, Decasper98 cannot anticipate Implicit’s inventions. Implicit, however, does not rely on its

⁹ The Court recognizes that in the reexaminations the PTO is considering validity under a preponderance of the evidence standard, as opposed to the clear and convincing evidence standard that applies here. The reasoning of the PTO, however, is persuasive and supports the Court’s conclusions.

¹⁰ The examiner in the ‘163 Patent ACP also rejected Claim 1 as obvious in light of Decasper98. ‘163 Patent ACP at 93-117.

¹¹ In the alternative, the examiner in the ‘857 Patent ACP rejected Claims 1, 4 and 10 as obvious under Decasper98, as well as obvious under Decasper in view of IBM96 and IBM96 and Nelson combined. *Id.*, at 19, 22, 31.

¹² As the parties explained in the tutorial IP (Internet Protocol) is responsible for addressing hosts and routing packets of messages across the internet. *See, e.g.*, Defendants’ Technology Tutorial at 10.

expert to contradict Dr. Calvert and Juniper’s assertions on these key points. Indeed, the Rebuttal Expert Report on Invalidity of Dr. Scott Nettles, Implicit’s expert, attempts to distinguish Decasper98 on different grounds, namely: (1) that Decasper98 does not dynamically identify components because components are initially identified by a static core; (2) plugins in Decasper98 are not components or software routines as defined by the Court; and (3) Decasper98 does not process packets so that input format matches output format. Nettles Invalidity Rebuttal Report ¶¶ 67-75. Implicit’s opposition arguments – Decasper98 does not convert data and fails to maintain state – are made without citation to expert support, although Nettles does assert that Decasper98 is a one-format system. *See, e.g.*, Nettles Invalidity Rebuttal Report, ¶ 64.

i. Effect of PTO Not Considering Decasper98 in Prior Proceedings

As an initial matter, Implicit disputes the impact – in terms of the deference the Court should provide to the presumption of validity of the patents – of the failure of the PTO to consider Decasper98 as prior art in the prosecution proceedings. Implicit argues that the failure of the PTO to consider Decasper98 does not lower Juniper’s burden to demonstrate invalidity by clear and convincing evidence under *Microsoft Corp. v. i4i Ltd. P’ship*, 131 S.Ct. at 2251, because the invention disclosed in Decasper98 is cumulative to prior art that was disclosed in the prosecution, namely Dietz, Method and Apparatus for Monitoring Traffic in a Network, United States Patent No. 6,651,099 (Dietz). *See* Invalidity Oppo. at 9. While Implicit attaches the Dietz patent to the Declaration of Spencer Hosie in Opposition to Invalidity (Hosie Invalid. Decl.), Implicit does not provide any analysis – by way of an expert or argument in its brief – as to *how* it believed Decasper98 was cumulative to Dietz. Instead, Implicit criticizes Juniper’s expert, Dr. Calvert, for failing to address Dietz. *Id.*

Dietz does not describe a router or gateway – the types of devices at issue here – but a “monitor and method of examining packets.” By contrast, Decasper98 teaches an architecture “to quickly and efficiently classify packets into flows, and to apply different policies to different flows” through “dynamically loaded” plug-ins. Dixon Decl., Ex. 1 at 2. Decasper98, on its face, closely matches the inventions covered by the patents in suit, while Dietz does not. Moreover, in his expert report Dr.

Calvert did address Dietz, if only to determine whether it, in combination with another reference (Kerr) rendered obvious the patents in suit. Calvert Invalidity Report, ¶¶318-331. Finally, in issuing the Office Actions and the Patent ACPs rejecting the asserted claims based in significant part on Decasper98, the PTO has apparently agreed that Decasper98 was a substantial new piece of prior art and *not* cumulative to Dietz, which had been before the PTO on its initial consideration. *See, e.g.*, MPEP § 2647 (requests for reexamination “will be denied if a substantial new question of patentability is not found based on patents or printed publication.”).

In sum, Implicit has not shown that the Dietz reference is cumulative to the aspects of Decasper98 that Juniper – and the PTO in the *inter partes* reexam– rely on. As such, Juniper faces a somewhat lighter burden under the clear and convincing evidence standard for demonstrating invalidity.

ii. Format Matching/Conversion

Implicit’s main argument against Decasper98 acting as an anticipating (or obviating) reference is that Decasper98 is a one format system – the packets that are processed are IP and the packets that come out of processing are IP. According to Implicit, the system covered by the patents is a system that *converts* data to ensure input and output formats match. Implicit and its expert acknowledge that – based on Implicit’s arguments during claim construction – the Court has already held that the “components” or software routines in Implicit’s system need not always “convert” data, but Implicit and Dr. Nettles contend that a system like Decasper98, which does not convert any data, cannot disclose the system covered by the patents. *See* Invalidity Oppo. at 13. Juniper responds that based on the Court’s claim construction, the components/software routines – the plug-ins in Decasper98 – need not actually convert anything; simple processing is enough. *See* Claim Construction Order [Docket No. 69 in Case No. 10-4234] at 9 (“‘conversion’ is not a necessary limitation with respect to the processing of each packet”).¹³ Juniper also contends, supported by Dr. Calvert, that Decasper98 discloses the possibility

¹³ In issuing its ACPs the PTO likewise agreed that Implicit’s claims, as construed, can cover processing of data and need not convert data or formats. *See, e.g.*, ‘163 ACP at 16 (Decasper98 invention “can cover protocol stacks conforming to the OSI model, even where the same general type of processing is performed at each layer, and in a predetermined order”) and at 22 (“within IP, various plugins handle various different formatting needs”); ‘857 ACP at 15 (“There is no requirement of

1 of using a conversion plug-in, which – as Implicit’s expert agrees – is “conversion” pursuant to the
 2 patent claims. *See* Calvert Invalidity Report ¶ 553; Dixon Decl., Ex. 20 [Nettles Depo.] at 191:11-
 3 192:16.¹⁴

4 The Court concludes, based on clear and convincing evidence not contradicted by Implicit’s
 5 expert, that the system described in Decasper98 processes the IP packets in a manner that anticipates
 6 Implicit’s inventions.

8 **iii. State Information**

9 Implicit also argues that Decasper98 cannot anticipate or render obvious Implicit’s system
 10 because the Decasper98 system does not maintain “state” regarding each components’ processing of a
 11 specific message, as required by the claims. Implicit argues that as a basic IP router, Decasper98 is not
 12 guaranteed to even see all packets of a particular message and, therefore, cannot manage the processing
 13 and maintenance of all packets of a particular message or flow, which is essential to the ‘163 Patent.
 14 *See* Oppo. at 9-10, 13. Relatedly, Implicit argues that given its structure, Decasper98 maintains only
 15 “soft state” information and not “hard state” as required by the ‘163 Patent to ensure all packets in a
 16 message are processed consistent with the dynamically identified path. Soft state, according to Implicit,
 17 means that the state information could be dumped at any time, jeopardizing the ability of a system to
 18 fully process all packets in a flow that, for example, contains TCP data. Oppo. at 14.

19 Juniper responds that because Decasper98 explicitly indicates it can be implemented as a router
 20 at the edge of a network, it would – in that configuration – see every message in a particular flow. *See*
 21 Decasper98 at 2; Juniper Reply on Invalidity at 7. Implicit did not address or rebut this point during oral
 22 argument. Juniper also notes that Implicit has no evidence that Decasper98 cannot maintain “hard” state
 23 and that Implicit’s citation to portions of Decasper98 that allow an administrator to “reset state

24 _____
 25 multiple formats or any conversion of format in the claims”) and at 17 (nonetheless evidence showed
 26 that plugins in Decasper98 have differences in format, “which would require coordination of formats
 between the plug-ins.”).

27 ¹⁴ Juniper also notes that the patents’ inventor, Edward Balassanian, testified that an all IP
 28 processing system could satisfy the input/output matching limitations. Dixon Decl., Ex. 34 [Balassanian
 Rule 30(b)(6) deposition] at 1054:14-24.

information,” *i.e.*, dump it when useful, does not undermine the fact that Decasper98 maintains state for the flows it processes. Juniper Reply on Invalidity at 7-8 & n.8 (noting that Decasper98 explains that the administrator can set the system to clear out or recycle cached entries in a flow table as necessary, but also the system is capable of exponential expansion to handle more cached records, demonstrating an ability to keep “hard state”); *see also* Decasper98 at 5, 9. The Court agrees with Juniper’s evidence that Decasper98 demonstrates the “maintenance of state” limitation of the patents.

As such, the Court finds that based on clear and convincing evidence, which was not before the PTO initially, Decasper98 anticipates the asserted claims.

iv. Obviousness as Single Prior Art Reference

Having found that Decasper98 anticipates the asserted claims, the Court need not reach whether Decasper98 renders the asserted claims obvious. The Court notes, however, that Implicit’s argument against Decasper98 as an obviousness reference is based on its “meta-level” argument that Decasper98 cannot anticipate because it is a single-format basic router. As discussed above, the Court has found that Decasper98 explicitly contemplates use as an edge-router, contemplates and ensures input and output format matching within the parameters of IP processing, and likewise contemplates the use of a conversion “plug-in” which could convert data (if that limitation were required, although under the Court’s Claim Construction Order, it is not). Those findings support a conclusion that Implicit’s claims are unpatentable over (as rendered obvious by) Decasper98.

C. Whether Decasper98 in Combination with IBM96 and Nelson Render the Asserted Claims Obvious

The Court has found that Decasper98 alone both anticipates the ‘163 patent and renders it obvious. Further, the Court finds that Decasper98 in combination with IBM96 and Nelson render the asserted claims obvious.

IBM96 and Nelson disclose the use of software components (plugins) for data compression and/or decompression, that according to Juniper and Dr. Calvert, one skilled in the art could have appreciated applying within the Decasper98 system. *See* Calvert Invalidity Report, ¶¶ 673-700. Dr.

Calvert explains that IBM96 teaches compression as a useful technique for gateways and routers, and that it would have been obvious for one skilled in the art to use IBM's stateful compression component with the "extensible" Decasper98 router. *Id.*, ¶¶ 674, 675, 678, 684. Dr. Calvert contends that Nelson, which further elaborates on specific compression algorithms that can be used to implement a "stateful" compression plugin, would likewise have been obvious to use with Decasper98's flexible and expandable router architecture. *Id.*, ¶¶ 689, 691.¹⁵

Implicit responds that the IBM96 and Nelson references cannot combine with Decasper98 to render the asserted claims obvious because IBM96 and Nelson speak to "stateful compression" of data. Compression of data does not alter its format, and so these references cannot disclose the conversion aspects of Implicit's patents. However, as noted above, the Court has rejected the requirement that each packet be converted. Implicit also argues that a stateful compression algorithm would require all packets of a message to be maintained together, something which Decasper98 as a "node router" couldn't do. However, as noted above, Decasper98 explicitly disclosed its use as an "edge router," which could maintain the packets. Finally, Implicit relies on Dr. Nettles' conclusion that there would be no motive to combine Decasper98 with IBM96 and/or Nelson because the compression techniques discussed in IBM96 and Nelson are not suitable for IP routers. Nettles Invalidity Rebuttal Report, ¶¶ 102, 108. Dr. Nettles does not explain in any detail why the IP router described by Decasper98 would not function with or could not be reasonably combined with IBM96 – which contemplates use of compression algorithm with a router – and/or Nelson, and the argument is not fleshed out in Implicit's brief. *See Invalidity Oppo.* at 15 (stating simply that "[n]or would there be motive to combine Decasper and the compression combinations," citing to Nettles Invalidity Rebuttal Report, ¶ 102). Dr. Calvert's

¹⁵ In the '857 ACP, the examiner rejected Claims 1, 4 and 10 in light of Decasper98 in view of IBM96, in light of Decasper98 teaching use of router plugins and IBM96 teaching a router platform having a compression algorithm. The examiner found that combining the two would have been obvious in an effort to achieve faster data transmission and processing. '857 ACP at 22-23 (citing Nelson to illustrate features of a stateful compression algorithm). In light of the extensible nature of Decasper98, IBM96's teaching use of a compression algorithm in a router, and the higher data rates, improved response times, and lower costs from compression in transmitting data, the examiner found explicit reasons as to why the references would be combined by one skilled in the art. *Id.*, at 24; *see also id.* at 35 (noting LZ compression algorithm discussed in Nelson applied in the router architecture of IBM96 provides the basis to add it as a Decasper98 plugin).

1 discussions on reasonableness and motive for those skilled in the art for combining Decasper98 and
2 IBM96 and/or Nelson are much more thorough, and stand essentially un rebutted.

3
4 **D. Objective Evidence of Nonobviousness**

5 Finally, Implicit can challenge Juniper's obviousness argument by showing that the "commercial
6 success" of its inventions covered by the patents "supports its contention of nonobviousness." *Demaco*
7 *Corp. v. F. Von Langsdorff Licensing, Ltd.*, 851 F.2d 1387, 1392 (Fed. Cir. 1988). In order to defeat
8 a showing of obviousness on summary judgment, a patentee must make a "prima facie case" showing
9 "both that there is commercial success, and that the thing (product or method) that is commercially
10 successful is the invention disclosed and claimed in the patent." *Id.*, at 1392. Implicit relies on its
11 development and sales contracts with Intel, AMD and Raytheon. Invalidity Oppo. at 7 & 16. In
12 particular, Implicit relies on an agreement it entered into with Intel to develop products (which never
13 made it to market), Implicit's discussion with Sprint about an Implicit prototype, and an "approach"
14 about investment made by a venture capital firm. Invalidity Oppo. at 7-8.

15 Juniper responds that Implicit has failed to show that any of its commercial embodiments
16 actually satisfy the asserted claims' elements, and that the development agreements and inquiries
17 Implicit relies on do not show any true "commercial success." The Court agrees with Juniper.

18 While Implicit mentions contracts with Intel, AMD and Raytheon, Implicit does not cite to
19 copies of those contracts or provide expert analysis explaining that the use of the asserted patents'
20 technology was key to those contracts. (*See, e.g.*, Calvert Invalidity Report, ¶ 775 (noting no evidence
21 that contracts with Intel and AMD embodied technology of patents in suit and citing Balassanian
22 deposition admitting the Raytheon contract had nothing to do with the patents-in-suit)). Implicit relies
23 on the fact that Intel's due diligence engineer testified that Implicit's technology was novel, and that
24 Intel entered into agreements with Implicit to develop an early version of an Ipad-type device and digital
25 music player, and that Intel's venture capital business "approached" Implicit about investing in the
26 company in late 2001. Oppo. at 7. However, there is no evidence before the Court that the technology
27 covered by the contracts with Intel or the technology discussed by the engineer in his deposition
28

1 included the technology at issue in the asserted patents. *See* Ex. A to the Hosie Decl., at 61-68; *see also*
2 Calvert Invalidity Report, ¶ 778.¹⁶

3 Implicit also relies on the conclusion of Sprint, which met with Implicit's predecessor company
4 BeComm in early February 1999. Oppo at. 16. According to a document created around that time by
5 Sprint, Sprint considered the BeComm communication platform prototype (Portal) to be unique,
6 exciting, and a good strategic opportunity for Sprint. *See* Hosie Decl., Ex. G at SP-175-0731. Sprint,
7 however, made that assessment "based on information to date and assuming further assessment is
8 positive." *Id.* There is no evidence that Sprint's initial reactions, based on two meetings on one day,
9 were supported by further assessment. An initial reaction based on a preliminary exposure cannot show
10 that Implicit achieved "commercial success" or was addressing a long felt but unresolved need.¹⁷

11 For the foregoing reasons, the Court finds – based on clear and convincing evidence presented
12 by Juniper's expert – that the asserted claims in the '163 and '857 patents are anticipated as well as
13 rendered obvious by Decasper98 alone and/or in combination with IBM96 and Nelson. The conclusion
14 is supported by the findings of the Patent Office ACPs. The Court also finds that in light of the strong
15 evidence of obviousness, Implicit's weak evidence of secondary of non-obviousness cannot overcome
16 that conclusion.

17 Concluding that the asserted claims of the patents are invalid, the Court need not reach whether
18 Juniper and F5 are entitled to summary judgment on non-infringement. However, as discussed below,
19 as alternate grounds to granting the motion for summary judgment as to invalidity, the Court finds that
20 defendants are entitled to summary judgment on non-infringement as well.

21
22
23 ¹⁶ Moreover, even if the Implicit technology Intel was seeking to secure through its contract with
24 Implicit was the technology of the asserted claims, Implicit has cited no evidence that any of the projects
25 Implicit worked on for Intel were released to market or that the duration or terms of the Intel contracts
amounted to commercial success. *See* Calvert Invalidity Report, ¶ 775.

26 ¹⁷ The PTO noted that Mr. Balassanian did not demonstrate that any commercial success of his
27 company (presumably the contracts with Intel, AMD and Raytheon) was due to the claimed features of
28 the '857 patent. '857 ACP at 44. The PTO also viewed skeptically Implicit's assertion that commercial
success was shown by numerous licenses Implicit had negotiated with other third-parties for use of the
technology, because it was "reasonable to assume the licenses were taken in lieu of defending []
infringement suits." *Id.* at 46-47.

II. Juniper's Motion for Summary Judgment on Non-infringement

Juniper moves for summary judgment arguing that Implicit has failed in its burden to establish a material issue of fact on its infringement claims. Juniper rests its argument on the fact that Implicit's expert did not consider the code for the actual accused products – the SRX and J series – and as a result had to extrapolate as to how the SRX and J series products work from generalities present in the operating code (JUNOS) that both products use. That “high level” assumption, according to Juniper, cannot carry Implicit's burden in opposing summary judgment. For the reasons discussed below, the Court agrees and rejects Implicit's attempt to shift its own burden – to show a material question of fact supporting infringement in this technically complex case with highly specific claim limitations – onto Juniper. *L & W, Inc. v. Shertech, Inc.*, 471 F.3d 1311, 1318 (Fed. Cir. 2006).

A. Legal Standard

To find infringement, “the court must determine that every claim limitation is found in the accused device.” *Playtex Products, Inc. v. Procter & Gamble Co.*, 400 F.3d 901, 909 (Fed. Cir. 2005) (internal citations omitted). Summary judgment of non-infringement is a two-step analysis. First, the claims of the patent must be construed to determine their scope, as a question of law. *Pitney Bowes, Inc. v. Hewlett-Packard Co.*, 182 F.3d 1298, 1304 (Fed. Cir. 1999) (internal citation omitted). Second, “a determination must be made as to whether the properly construed claims read on the accused device.” *Id.* The determination of infringement is generally a question of fact. *Lockheed Martin Corp. v. Space Sys./Loral, Inc.*, 324 F.3d 1308, 1318 (Fed. Cir. 2003). Since the ultimate burden of proving infringement rests with the patentee, an accused infringer may establish that summary judgment is proper “either by providing evidence that would preclude a finding of infringement, or by showing that the evidence on file fails to establish a material issue of fact essential to the patentee's case.” *Novartis Corp. v. Ben Venue Labs., Inc.*, 271 F.3d 1043, 1046 (Fed. Cir. 2001).

B. Motion to Exclude Expert Testimony of Dr. Alexander

As an initial matter, Implicit moves to exclude the expert testimony of Juniper's non-

1 infringement expert, Dr. Peter Alexander. Juniper opposes that motion and, in response, cross-moves
2 to exclude the expert testimony of Implicit's infringement expert, Dr. Scott Nettles. Implicit attacks Dr.
3 Alexander's testimony on two grounds. The first ground is that Dr. Alexander did not independently
4 explore how the accused products operate and, therefore, his opinions are baseless. The second ground
5 is that when Dr. Alexander testified that Implicit's expert Dr. Scott Nettles failed to prove by a
6 "preponderance of the evidence" that Juniper infringed, Dr. Alexander is not only incorrect, he is
7 attempting to usurp the jury's function.

8 Federal Rule of Evidence 702 provides that expert testimony is admissible if "scientific,
9 technical, or other specialized knowledge will assist the trier of fact to understand the evidence or to
10 determine a fact in issue." Fed. R. Evid. 702. Expert testimony under Rule 702 must be both relevant
11 and reliable. *Daubert v. Merrell Dow Pharms.*, 509 U.S. 579, 589 (1993). When considering evidence
12 proffered under Rule 702, the trial court must act as a "gatekeeper" by making a preliminary
13 determination that the expert's proposed testimony is reliable. *Elsayed Mukhtar v. Cal. State Univ.*, 299
14 F.3d 1053, 1063 (9th Cir. 2002), *amended by* 319 F.3d 1073 (9th Cir. 2003).

15 Regarding Implicit's first argument, the Court finds that given the scope of Dr. Alexander's role
16 – which was to rebut Implicit's expert's infringement analysis – Dr. Alexander was not required to
17 conduct the in-depth "infringement analysis" Implicit contends he should have undertaken. Moreover,
18 Implicit's arguments do not establish that Dr. Alexander's opinions are unreliable such that they should
19 be excluded, but go more to the weight a jury may give to Dr. Alexander's opinion (assuming Dr.
20 Alexander's understanding of the accused products is relevant to his assertion that Dr. Nettles' opinions
21 are deficient). With respect to the second argument, the Court finds that Implicit's contentions
22 regarding the deficiencies in Dr. Alexander's testimony (*i.e.*, how Dr. Alexander criticized Dr. Nettles'
23 conclusions and the basis for those criticisms) go to the weight a jury might give Dr. Alexander's
24 conclusions and not their admissibility. *See, e.g., Daubert*, 509 U.S. at 596 ("Vigorous
25 cross-examination, presentation of contrary evidence, and careful instruction on the burden of proof are
26 the traditional and appropriate means of attacking shaky but admissible evidence."). Finally, the fact
27 that Dr. Alexander opines that Dr. Nettles has not established infringement by a preponderance of the
28

1 evidence, is not a basis on which to exclude his testimony. *See, e.g., Symbol Technologies, Inc. v.*
2 *Opticon, Inc.*, 935 F.2d 1569, 1575 (Fed. Cir. 1991) (“testimony on the ultimate issue of infringement
3 is permissible in patent cases”).

4 Implicit’s Motion to Exclude the testimony of Dr. Alexander, therefore, is DENIED. As
5 Juniper’s cross-motion to exclude Dr. Nettles’ report is made only in the alternative to its opposition
6 to Implicit’s motion to exclude Dr. Alexander’s testimony, that motion is likewise DENIED.

7 8 **C. Non-infringement**

9 Juniper argues that Implicit has failed to produce any evidence that the accused products – the
10 SRX and J-Series gateways and routers – contain all of the claim limitations. Juniper argues that by
11 relying only on an analysis of source code from a product series that is not alleged to infringe (the
12 “Multiservices” products) , and by relying on general technical and marketing documents regarding the
13 JUNOS operating system that is used across Juniper products, Implicit has failed to prove that the SRX
14 and J series products infringe.

15 Implicit does not contest that its expert, Pavel Treskunov, limited his code analysis to the
16 Multiservices products, as well as plug-ins/software components used by the Multiservices products.
17 Nor does Implicit contest that its infringement expert, Dr. Scott Nettles, relied on the Treskunov code
18 analysis to create his expert report regarding infringement. Indeed, throughout his report Dr. Nettles
19 relies on the Multiservices code and a list of plugins for the Multiservices products. *See, e.g.,* Nettles
20 Expert Infringement Report, Ex. A [Ex. 8 to Hefazi Decl.], ¶ 14 (citing to Exhibit 3 for list of plug-ins
21 related to flow based processing); ¶ 32 (citing code). But *nowhere* in his discussion does Dr. Nettles
22 acknowledge that the code and plugins are for the Multiservices products. Nor does Dr. Nettles explain
23 why *he believes* that the Multiservices code and plugins he analyzes are used in the SRX or J-series
24 products, so that the SRX and J-series products would read on each limitation of the asserted claims.
25 Relatedly, as Juniper’s non-infringement expert Dr. Alexander points out, when Dr. Nettles conducts
26 his limitation-by-limitation infringement analysis, Dr. Nettles does not distinguish between different
27 types of SRX products (branch vs. high end) or between the SRX and J series products. *See* Alexander
28

1 Report, ¶ 71.¹⁸

2 Dr. Alexander, on the other hand, explains that Nettles' reliance on the CPCD plugin¹⁹ is
3 misplaced, as that plugin is not used in the SRX or J series products. Alexander Report, ¶¶ 75-83. Dr.
4 Nettles' reliance on the "mspman" packet handling code, which is related to the Multiservices PIC
5 Manager,²⁰ is also misplaced as Dr. Alexander concludes the Multiservices PIC add-on card is a product
6 for routers that is not used by the SRX or J series products. Alexander Report, ¶¶ 84-96. Dr. Alexander
7 also criticizes Dr. Nettles' reliance on the "service sets" concept, which Dr. Nettles uses as the basis for
8 his flow-based processing analysis and of the "dynamically identifying a non-predefined sequence of
9 components" limitations. Alexander Report, ¶¶ 87. Dr. Alexander concludes that the Service Set was
10 designed for use with the Multiservices products and is not used in the accused products. *Id.*; *see also*
11 *id.*, ¶¶ 88-97.

12 Implicit argues that it can rely on Treskunov's and Nettles' analysis under the "proxy" or
13 exemplar infringement theory because the Multiservices products as well as the SRX and J series
14 products all use the One JUNOS Operating System. Therefore, Implicit argues that all of the products
15 can be assumed to operate and infringe in the way Dr. Nettles asserts that Multiservices products
16 infringe. However, Implicit provides no legal authority that would allow a device that is not accused
17 of infringing to be the proxy for accused devices.²¹

18
19
20 ¹⁸ As a specific example *see* Dr. Nettles's evidentiary support for asserting SRX and J series
21 products read on Claim 1a, where Dr. Nettles does not cite any documents specifically regarding the J
22 series products. Nettles Infringement Report, ¶ 71 (citing evidence, including Multiservices plugins as
support for meeting Claim 1a limitation), ¶ 73 (citing *Junos Security* document referencing SRX but not
J series products as support for meeting Claim 1a limitation).

23 ¹⁹ *See* Nettles Infringement Report at 37 (example of "CPCD plugin handling first packet"); at
24 51 ("example of CPCD plugin returning 'session ignore' code as a result of the processing of the first
packet of a session").

25 ²⁰ *See* Nettles Infringement Report at 20 (discussing the mspman daemon).

26 ²¹ Under the "proxy" theory, an expert who analyzes only one device in depth can testify that
27 other accused products operate similarly as to the specific claim at issue. *See, e.g., Tivo, Inc. v.*
28 *Echostar Communications Corp.*, 516 F.3d 1290, 1308 (Fed. Cir. 2008) ("there is nothing improper
about an expert testifying in detail about a particular device and then stating that the same analysis
applies to other allegedly infringing devices that operate similarly, without discussing each type of
device in detail.").

More importantly, Dr. Nettles provides no reasoned opinion showing that the SRX and J series products *actually* operate in the same way – for purposes of infringement analysis – as the Multiservices products. A patentee cannot “simply ‘assume’ that all of [accused infringer’s] products are like the ones that [the patentee] tested and thereby shift to [the accused infringer] the burden to show that is not the case.” *L & W, Inc. v. Shertech, Inc.*, 471 F.3d 1311, 1318 (Fed. Cir. 2006). As Dr. Alexander points out, having a single operating system “does not mean [] that all of that software is loaded on each and every system.” Tavakoli Depo. at 151:4-19; *see also* Alexander Report, ¶ 74.²² Different products have different functionality enabled and use different portions of the JUNOS code tree. Tavakoli Depo. at 151:16-19.

To support its argument that the Multiservices products operate in the same way – for purposes of infringement – as the SRX and J series products, Implicit relies on the deposition testimony of Krishna Narayanaswamy, a Juniper deponent. In his testimony, Mr. Narayanaswamy agreed that the Multiservice DPC used “Junos-based” “flow-based” processing as did the J series routers and the SRX products. Narayanaswamy Depo. [Exhibit B to Hosie Declaration re Juniper Infringement] at 190:16 - 192:3. However, that testimony does not establish that the “flow-based” processing being discussed in general would read on the very specific claim limitations in this case.²³ Therefore, while Implicit has shown – by citation to deposition testimony and Juniper’s technical documents – that Juniper products use JUNOS as the base source code (*see, e.g.*, Appendix A to Expert Report on Infringement of Dr. Scott Nettles at 7 (citing JUNOS OS: The Power of One Operating System at 1)), Implicit has *not* shown how the accused products use specifically identifiable portions of JUNOS to support a finding that the accused products read on every element of the asserted claims.

²² For example, Dr. Alexander points out that Dr. Nettles cites to JUNOS source code allowing for both flow-based and packet-based processing. Nettles Infringement Report, ¶¶ 31-32. According to Dr. Alexander, Dr. Nettles further contends that the high-end SRX series *cannot* perform packet-based processing. Alexander Report, ¶ 86. The fact that one of the SRX products at issue does *not* perform a function that is enabled by JUNOS shows the weakness of Implicit’s argument. Simply demonstrating that all products draw code blocks from JUNOS cannot support showing of infringement here, especially considering the differences between the accused products.

²³ Mr. Narayanaswamy also testified that other Juniper products were “flow-based” but did not operate on JUNOS. *Id.*, at 192.

That missing connection cannot, on this summary judgment record, be made based on the fact that all of Juniper's products – or at least the SRX, J and Multiservices products – operate “flow based” processing. As Implicit's own founder testified, the patents at issue do not automatically cover “flow based” processing systems. Balassanian Depo. at 985:5-9 (Ex. B. to Hefazi Reply Decl.[Docket No. 191-3]). Rather, specific elements of the patent claims at issue – *e.g.*, dynamically identifying components to process the data after receipt of the first packet, maintain state specific to a software routine for a specific message that is not information related to an overall path – must be shown to be practiced within the “flow based” processing used by the SRX and J series products.²⁴ Implicit's evidence simply fails to make a showing sufficient to create a material issue of fact to defeat Juniper's summary judgment motion.

Finally, Implicit repeatedly criticizes Juniper for failing to rebut Implicit and Dr. Nettles' high-level generalized arguments that the One JUNOS source code allows for an inference that the accused products operate similarly to the actually-analyzed Multiservices products. However, Implicit is impermissibly attempting to shift its own burden – to show a material question of fact supporting Juniper's infringement – onto Juniper. *L & W, Inc. v. Shertech, Inc.*, 471 F.3d at 1318. Implicit has the burden, especially in a technically complex case such as this, to show by expert testimony how the accused products actually work. *See, e.g., Centricut, LLC v. Esab Group, Inc.*, 390 F.3d 1361, 1370 (Fed. Cir. 2004) (“the patentee cannot satisfy its burden of proof by relying on testimony from those who are admittedly not an expert in the field.”). Dr. Nettles' assertions about JUNOS are far too generalized and rely too much on technical citations and code for products other than the SRX and J series, to support a claim-by-claim and limitation-by-limitation infringement analysis necessary to meet Implicit's burden in opposing summary judgment. *See, e.g., Eugene Baratto, LLC v. Brushstrokes Fine Art, Inc.*, 701 F. Supp. 2d 1068, 1080-81 (W.D. Wis. March 24, 2010) (exemplar approach “does not

²⁴ Given the complex nature of the products at issue, and the specific, technical requirements of the asserted claims (*e.g.*, maintaining component-based state information that is not related to the overall processing path), “even slight variations in product functionality may easily be the difference between infringement and non-infringement.” Alexander Report, ¶ 74. As such, even assuming that portions of the JUNOS code may practice on the method claimed, there still needs to be evidence showing not only that the accused products utilize the infringing aspects of the JUNOS code but how that use of that code in the *accused* products infringes. That showing has not been made here.

work when the only evidence supporting such grouping is merely that different versions of a product are part of the same family and have the same basic functionality.” *Baratto v. Brushstrokes Fine Art, Inc.*, 701 F. Supp. 2d 1068, 1081 (W.D. Wis. 2010) (internal quotation omitted)).

The Court concludes, therefore, that Implicit has failed to raise a material question of fact on infringement against Juniper, further supporting the grant of Summary Judgment in Juniper’s favor.²⁵

III. F5’s Motion for Summary Judgment on Non-infringement

F5 argues that its products do not meet the “dynamically identifying” and “non-predefined sequence” post-first packet limitations because the accused F5 “BIG-IP” products use sequences of components (Hudfilters) that have been fully identified and arranged (in Hudchains) prior to any data entering the system. Implicit responds that the BIG-IP products infringe because the Hudchains are simply processing policies, and F5’s products create a “stateful” message-specific path (the Connflow) *after* the first packet of a message enters the system. It is the creation of the Connflow, Implicit argues, that satisfies the dynamic identification of a non-predefined sequence of components limitation. For the reasons discussed below, the Court agrees with F5, and finds Implicit has not raised a material issue of fact as to the “dynamically identifying” and “non-predefined sequence of components” limitations.²⁶

A. The Technology of the Patents and F5’s BIG-IP Products

As discussed in the Claim Construction Order, according to Implicit’s complaint, the heart of

²⁵ Juniper also argues, as an alternate ground for summary judgment of non-infringement, that the source code and other materials relied upon by Implicit and Dr. Nettles fail to show that both of the accused products meet three other limitations: (1) that they “select individual components” in a compatibility check; (2) that the plurality of components selected meet the limitations requiring them to keep specific pieces of state; and (3) that the components are “dynamically identified” after the first packet is received. Because the Court has granted summary judgment of non-infringement, the Court need not reach these arguments but notes that Juniper has made a strong case that its products do not practice these three limitations.

²⁶ F5 also argues that the accused products do not read on the “for each of a plurality of packets of the message in sequence,” “state information” and selecting components based on a “message” limitations. However, given the Court’s findings set out in this order, it need not reach F5’s additional arguments. *See Kahn v. GMC*, 135 F.3d 1472, 1477 (Fed. Cir. 1998) (“The absence of even a single limitation of [a claim] from the accused device precludes a finding of literal infringement.”).

1 the patents' invention is a networking process where "discrete computer function[s], e.g., processing
2 http server requests over TCP/IP, streaming a video web-based client, or managing voice-over-ip calls,
3 would be built into a discrete software module, called a 'bead.'" The system devised could
4 "dynamically" "receive a stream of data – say video – determine what services were necessary to render
5 that content and where the content was to be rendered, and then assemble – or string together – the
6 requisite service beads (modules) at run-time." See, e.g., FAC [Docket No. 31, Case No. 10-3365], ¶
7 16. This system, according to Implicit, dramatically departed from the prior art where a developer of
8 applications had to anticipate who would use the applications and for which devices and content, and
9 then build-in the ability to handle the anticipated demands. *Id.*, ¶ 11. The prior art model had many
10 short falls, including an "ever-increasing complexity, cost, and processing overhead . . . Given that all
11 anticipated uses had to be preconfigured at build-time, any unanticipated new use, e.g., a different
12 format or a different device, would simply break the system. The developer had to have the foresight
13 to specify explicitly all possible configurations in advance, a difficult task in a rapidly changing world."
14 *Id.*, ¶ 12.

15 The '163 Patent entered reexamination in 2008 and emerged in June 2010 with additional
16 limitations. The purpose and result of the reexamination was to distinguish the '163 series of patents
17 from the prior art, and specifically from David Mosberger, "Scout: A Path-Based Operating System,"
18 Doctoral Dissertation Submitted to the University of Arizona. See Claim Construction Order at 3. The
19 '163 reexamination added a number of significant limitations in order to distinguish Mosberger,
20 including "*dynamically identifying a non-predefined sequence of components*" for processing
21 "messages," wherein "*dynamically identifying includes selecting individual components to create the*
22 *non-predefined sequence of components. . . .*" '163 Reexam.

23 As discussed in the Claim Construction Order, Implicit's September 1, 2009 Amendment and
24 Response in the '163 Reexam cited to the '163 specification distinguishing prior art systems which
25 "typically use predefined configuration information" to load the correct series of conversion routines
26 that make up the "path." 9/1/09 Amendment and Response at 18. Implicit was attempting to distinguish
27 itself from prior art which had to "pre-bake" complete processing sequences for all potential formats
28

1 of data a system might want to process at compile time; Implicit patented inventions which require that
2 the sequences of components to be used to process messages be *dynamically* identified during run-time
3 but post-first packet, in order to accommodate for new data formats or different devices. *See, e.g.*, FAC,
4 ¶ 12. Implicit’s Amendment and Response makes clear that what it was disclaiming in the prior art was
5 use of preconfigured sequences of routines, in other words preconfigured processing paths. *See* 9/1/09
6 Amendment and Response at 18 (“[T]he sequence of conversion routines (or ‘path’) is not configured
7 prior to receiving the first packet of a message.”); *see also* 2/8/10 Amendment and Response at 16.

8 The accused F5 products are all part of the BIG-IP line of products. The BIG-IP products are
9 application controllers that allow clients to manage, inspect, transform and route their application traffic.
10 Declaration of Dr. James Storer, ¶ 42. For example, an internet users’ application request (*e.g.*, placing
11 an order with an e-retailer, seeking access to bank account information) travels over the internet to an
12 F5 customer’s BIG-IP, which then processes and forwards the request to specific servers on the client-
13 side. *Id.*, ¶ 43. Given the massive amounts of data being processed per second, the BIG-IP products
14 are designed to help increase the speed and performance of the client websites. *Id.* ¶ 41.

15 The BIG-IP products use a common operating system, Traffic Management Operating System
16 (TMOS). TMOS is a “modular,” or “pluggable” operating system. *Id.* ¶ 50. Within that system are
17 modules known as Hudfilters, which are code units that perform a processing step on the network traffic.
18 *Id.* ¶ 53. Hudchains are comprised of an identified set of Hudfilters that have been interconnected to
19 form a data processing sequence. *Id.* ¶ 54. The contents of the Hudchains – Hudfilters connected in a
20 specific sequence – are determined by Profiles selected by clients within the boundaries of F5’s hard-
21 coded configuration rules, called Hudfilter constraints. *Id.*, ¶ 56. BIG-IP ships with a set of Profiles
22 that control the behavior and processing of particular types of network traffic, and client system
23 administrators can select which Profiles to implement or can build custom Profiles (for example, client
24 administrators can decide to enable cookie encryption for HTTP profiles). *Id.* Once the system
25 administrator selects or creates the Profiles, the BIG-IP system will use a Hudchain template to create
26 the Hudchains that include all necessary Hudfilters to process the client’s traffic in light of the client’s
27 needs. *Id.*, ¶ 64. The Hudchain template is then built, using a Hudchain_select command, which
28

1 ensures compatibility between the Hudfilters. *Id.*, ¶¶ 65-66. At that point, the system constructs the
 2 actual Hudchain by implementing the Hudchain_construct function. *Id.*, ¶ 67. The system will then
 3 assign the Hudchain a “listener” pointer which, when a particular type of traffic arrives, will direct the
 4 packets to the Hudfilters specified in the Hudchain for processing that type of message. *Id.*, ¶¶ 68, 69.
 5 As such, the Hudchains are fully identified and determined before any network traffic enters the system.
 6 *Id.*, ¶ 69.

7 Once the first packet of a message or flow hits the system and a TCP connection is established
 8 – which requires the exchange of information between the user computer and the client server and
 9 therefore occurs only after the exchange of three preliminary packets of a message/flow²⁷ – the system
 10 creates a Connflow. The Connflow is a dedicated area of memory in the system, that has a pointer to
 11 the first Hudfilter of an existing Hudchain to allow the processing of the flow. *Id.*, ¶70; *see also id.*
 12 ¶¶86-87. The Connflow stores information unique to a particular flow and tracks, for example, the
 13 Hudfilter’s connection state and the expected sequence number of the next packet of the flow. *Id.*, ¶ 87.
 14 However, the individual Hudfilters processing the packets as well as the Hudchains setting out the
 15 parameters for the processing, are stored in separate memory space and are loaded into the system on
 16 initialization. *Id.* Once a message/flow starts to be processed, any changes in the Hudchain or additions
 17 to the functionality of the Hudfilters, will not be applied to any existing flow.²⁸ Those changes will only
 18 be applied to new flows that enter the system after the revised or new Hudchain is constructed. *Id.*, ¶
 19 70.

20 Implicit and its expert Dr. Scott Nettles do not dispute that this is the basic design and
 21 configuration of the BIG-IP products. Instead, as discussed below, the main point of dispute between
 22 the parties is whether the Hudchain *or* the Connflow is the “sequence” or “path” that needs to be
 23 dynamically-identified by the system post-first packet in order to read on the claim limitations. Because
 24 the parties’ arguments hinge on claim construction – and largely reiterate arguments made during Claim
 25

26 ²⁷ *See id.*, ¶ 81.

27 ²⁸ For example, the system allows client administrators to implement iRules to enhance the
 28 functionality of specific Hudfilters. *Id.*, ¶ 71.

1 Construction – they lend themselves to resolution on summary judgment. *See, e.g., Rheox, Inc. v.*
2 *Entact, Inc.*, 276 F.3d 1319, 1324 (Fed. Cir. 2002).

3 4 **B. Non-Predefined Sequence of Components**

5 As noted above, the Court defined “non-predefined sequence of components” as “a sequence of
6 software routines that was not identified before the first packet of a message was received.” Claim
7 Construction Order at 6. In doing so, the Court noted that in the initial ‘163 Reexam, Implicit did not
8 disclaim the ability to rely in “some part” on predefined “configuration information” to identify the non-
9 predefined sequence, but disclaimed the use of “pre-configured paths.” *Id.* The Court also noted that
10 the claim language itself requires that identification of the sequence occur “after the first packet is
11 received.” *Id.*

12 F5 argues that its BIG-IP products cannot meet this limitation because the sequence of
13 components is pre-defined – it is defined in the Hudchains before any traffic hits the system. Implicit
14 counters that F5’s Connflows, created post-first packet, are the *actual* processing paths followed by the
15 packets in a message and that they represent the “sequence of components” identified “after the first
16 packet is received” which satisfy this limitation. The Court rejects Implicit’s argument. Implicit
17 attempts to add a requirement – that the method not only identify the sequence of components to be
18 used to process a particular type of message/flow but also “create” an actual, “stateful” path in memory
19 that will track the processing of one particular message/flow. *See* Oppo. at 13, 19.²⁹ The Court
20 recognizes the distinction between the Hudchains (which set out how types of messages/flows will be
21 processed) and the Connflows (which keep in memory the pointers to the initial Hudfilters and keep
22 track of the actual processing of the packets of a specific flow/message), but the identification and
23 selection of the individual components to form the processing sequence is what is required by the claim
24 language. That is what happens when the Hud_construct function creates the Hudchains based on the

25
26
27 ²⁹ Apparently, this is why Implicit proposed construing “create” as used in “wherein
28 dynamically identifying includes selecting individual components to create the non-predefined sequence
of components after the first packet is received” to mean “instantiate in memory.” Claim Construction
Order at 11. The Court rejected Implicit’s attempt and adopted the plain meaning of create. *Id.*

Policies adopted by the BIG-IP clients. This conclusion is consistent with Implicit's repeated characterization of its invention as one which was different from prior art because Implicit's patents identify the individual components needed process the message (the sequence) dynamically and post-first packet. *See, e.g.*, 9/1/09 Amendment and Response at 12 ("configuring a path at build time (i.e., identifying the components used to process message packets before actually receiving any message packets) is fundamentally different than configuring a path at run-time (i.e., identifying the components for processing message packets after receiving the packets). . . ."). It was this dynamic selection of components, occurring post-first packet, that would allow Implicit's system to accommodate a message carrying a new format or needing to be delivered to a new device.³⁰ FAC, ¶ 12.³¹

It is true that Implicit did not disavow using pre-configured processing information "in some part." *See* Claim Construction Order at 6. However, Implicit's *did* disclaim the use of a pre-configured sequence of components, as represented in the BIG-IP products as the Hudchain. *Id.* It is not disputed that the Hudchains which set out the processing routines to be applied to every authorized type of message/flow are defined and loaded into the system's memory prior to the arrival of the first packet. Therefore, they cannot read on this significant limitation.

C. Dynamically Identifying and Selecting Individual Components

In the Court's Claim Construction Order, the Court adopted the plain meaning of "dynamically identifying" because it was adequately defined in the wherein clause of the claim language ("wherein

³⁰ The undisputed fact that thousands of flows could use the same Hudchain at the same time but only one Connflow is created for each message/flow (*see* Oppo. at 12), does not change the Court's conclusion. The fact that Connflows are necessary to provide memory enabling and tracking the processing of each message/flow does not make them into a "non-predefined sequence of components." The Connflow is simply a feature that allows the identified, predefined sequence controlled by the Hudchain to run.

³¹ Implicit does not dispute that if the BIG-IP products receive a data packet that is *not* recognized or pre-authorized for processing by a specific Hudchain, the packet will not be processed. *See* Storer Report, ¶ 95; *see also id.*, ¶¶ 69, 153 (noting that in order to process high volumes of network traffic, BIG-IP uses preset processing sequences; a non-predetermined system using dynamic selection of processing routines "could not achieve the data throughput requirements of a large enterprise application."). BIG-IP, therefore, functions in a way that is contrary to the purpose of Implicit's inventions.

1 dynamically identifying includes selecting individual components to create the non-predefined sequence
2 of components after the first packet is received”). Claim Construction Order at 7. The Court also
3 construed “selecting individual components” as “selecting the individual software routines of the
4 sequence so that the input and output formats of the software routines are compatible.” *Id.*, at 11.

5
6 **i. Dynamically Identifying**

7 F5 argues that the BIG-IP products cannot meet the limitation of “dynamically identifying” a
8 non-predefined sequence, where “dynamically identifying includes selecting individual components to
9 create the non-predefined sequence of components after the first packet is received.” Claim
10 Construction Order at 7. F5 notes that no dynamic identification occurs post-first packet because the
11 Hudchains determine the components to be used for processing prior to any information hitting the
12 system. Implicit responds that because the actual processing of the packets is controlled by the
13 Connflow, which is not created until post-first packet, this limitation is met. However, the Connflow
14 is a structure in the memory used to track the actual processing of a specific message. By contrast, the
15 dynamic identification limitation is aimed at selecting the individual software routines, which in the
16 BIG-IP products happens through the predefined Hudchains prior to any traffic hitting the system. For
17 the same reasons as discussed above, the BIG-IP products cannot meet this limitation.³²

18
19 **ii. Selecting Individual Components**

20 The Claim Construction Order found that a “necessary” part of selecting individual components
21 is determining the compatibility between the output of one software routine and the input of the next.
22 Claim Construction Order at 11. F5 argues that Implicit has failed to identify a “compatibility check”
23 in the BIG-IP products that operates *during* the dynamic identification of the individual components
24 which, under Implicit’s argument, allegedly occurs during the instantiation of the Connflow. Instead,

25
26 ³² This analysis is not altered by calling the “non-predefined sequence of components” a “path,”
27 as Implicit’s expert, Dr. Nettles, does repeatedly. *See* Nettles Infringement Report, ¶¶ 53-59. The
28 claims do not require the dynamic selection of a physical processing path, they require the dynamic
selection of the full sequence of processing components. There is no support for Implicit’s attempt to
redefine the claims.

F5 argues that the compatibility check in the BIG-IP products is handled by the Hudfilter constraints implemented prior to the building of the actual, predefined Hudchains. Implicit responds that the claims do not require any particular “process” for ensuring compatibility, other than application of general policies (for example, industry standards) which ensure the packets of a message can be handed off for further processing from one component to the next. Implicit also argues that this “compatibility” can be ensured when components are matched together in creating F5’s Hudchains or Implicit’s LabelMapGet utility. Oppo. F5 at 24. However, like the “selecting of individual components” step of which it is a part, the compatibility step has to be done *post-first packet*. Implicit points to no evidence that there is any type of compatibility check in BIG-IP, other than in the Hudfilter constraints step run before the Hudchains are put in place and run well before the instantiation of the Connflow.³³ Moreover, whether or not Implicit’s LabelMapGet utility functions in a similar manner to the Hudfilter constraints used by BIG-IP products, there is no dispute that LabelMapGet is used in Implicit’s inventions post-first packet.³⁴

For these additional reasons, F5’s BIG-IP products do not read on necessary limitations in the claims and summary judgment should be granted in F5’s favor.³⁵

///

///

³³ See Nettles Deposition [Ex. C to Brun Decl.] at 158:23-159:6 (testifying that F5 satisfies the selecting individual components limitation by applying constraints before the first packet of a flow is received).

³⁴ Relatedly, although Implicit’s inventions may “rely[] in some part on ‘predefined configuration information’” in light of the LabelMapGet function used in a preferred embodiment, Implicit disclaimed use of a pre-defined sequence of processing components. Claim Construction Order at 6. The amended claim language and Implicit’s statements in the Reexam therefore require *some* dynamic-identification of *individual* components, ensuring the packets are compatible, post arrival of the first packet. Implicit has not pointed to any feature in the BIG-IP products that does this.

³⁵ Having determined that at least these significant limitations are not met by F5’s products, the Court need not address F5’s remaining non-infringement arguments.

CONCLUSION

For the foregoing reasons and for good cause shown, the Court GRANTS defendants' motion for summary judgment on invalidity. The Court also GRANTS both defendants' motions for summary judgment on non-infringement.

IT IS SO ORDERED.

Dated: March 13, 2013



SUSAN ILLSTON
United States District Judge